

What is claimed is:

1. A method for building a report, comprising the steps of:
  - a) defining a relational abstraction of a data store, the definition including a plurality of views, fields associated with the views, and any relations between the views;
  - b) selecting one of the views as a base view of a report; and
  - c) creating at least one report field using the views, fields and relations associated with the base view.
2. The method of claim 1 wherein the report field is created by:
  - a) specifying a relation path from the base view to a predetermined destination view, the relation path including zero or more relations;
  - b) if the specified relation path contains any to-many relations, creating an aggregate field by:
    - i) creating at least one nested field to be aggregated, where the relation path of the nested field begins at the destination view of the aggregate field ; and
    - ii) specifying an expression for aggregating the values associated with the nested field; and
  - c) if the specified relation path does not contain any to-many relations, creating a scalar field, by:
    - i) referencing a source field from the destination view; or
    - ii) creating an expression field that includes at least one nested field, where the relation path of the nested field begins at the destination view of the expression field.

3. The method of claim 1 wherein information returned in a report is limited by specifying a filter that limits any rows returned in the report.
4. The method of claim 2 wherein information returned in a report is limited by specifying a filter that limits any rows returned in the report.
5. The method of claim 2 wherein information returned in a report containing at least one aggregate field is limited by specifying a filter on the destination view associated with the aggregate field.
6. The method of claim 3 wherein a filter is specified that limits any rows returned in the report by:
  - 10 a) creating a plurality of fields for use as filter fields; and
  - b) specifying at least one logical filter operator and at least one corresponding filter value on each filter field.
7. The method of claim 4 wherein a filter is specified that limits any rows returned in the report by:
  - 15 a) creating a plurality of fields for use as filter fields; and
  - b) specifying at least one logical filter operator and at least one corresponding filter value on each filter field.
8. The method of claim 6 wherein a plurality of filter fields are organized into nested filter groups, filters are specified for each of the nested filter groups and the output of applying the filters of the nested filter groups are used to combine the results of the report.
- 20 9. The method of claim 7 wherein a plurality of filter fields are organized into nested filter groups, filters are specified for each of the nested filter groups and the output of applying the filters of the nested filter groups are used to combine the results of the report.

10. The method of claim 1 wherein results of the report are sorted by at least one field.

11. The method of claim 2 wherein results of the report are sorted by at least one field.

12. The method of claim 1 wherein results of the report are grouped by at least one field.

13. The method of claim 2 wherein results of the report are grouped by at least one field.

5 14. The method of claim 2 wherein a total aggregate field is created that provides a total  
of an aggregate field based upon relations contained in the relation path for the aggregate  
field.

15. The method of claim 2 wherein an SQL query that includes the report fields is  
generated, by::

10 a) generating an initial SELECT statement with a FROM clause that references the  
base view of the report;

b) adding each report field having an empty relation path to a first subsequent  
SELECT statement; and

c) adding each report field having a non-empty relation path to a second subsequent  
15 SELECT statement, by:

i) adding a JOIN clause from the base view of the report to an initial nested  
SELECT statement and a FROM clause referencing the destination view  
associated with a first relation in the relation path; and

ii) adding a subsequent nested SELECT statement inside the initial nested  
20 SELECT statement for each successive relation in the relation path.

16. The method of claim 15 wherein a report filter is added to an SQL query by adding a  
WHERE clause to an outer SELECT statement.

17. The method of claim 15 wherein a filter specified on a to-many relationship in the relation path of an aggregate field is added to an SQL query by adding a WHERE clause to the corresponding nested SELECT statement.

18. The method of claim 15 wherein an SQL query is optimized by consolidating the 5 nested SELECT statements associated with report fields where the relation paths are equal to or are a prefix of each other.

19. The method of claim 2 wherein a reusable field definition is created that includes any filters specified on to-many relationships in the relation path associated with a field.

20. The method of claim 2 wherein a reusable filter definition is created that includes 10 filter groups and fields associated with the report and operators and values that correspond to the filter groups and fields.

21. The method of claim 1 further comprising the steps of:

- a) defining a security principal;
- b) creating an access control entry for at least one element of information in the data 15 store, which access control entry grants or denies the security principal access to the element;
- c) defining a security filter on a view, by:
  - i) creating a filter that identifies at least one row in the view; and
  - ii) creating an access control entry denying the security principal access to any 20 information to be excluded by the filter; and
- d) enforcing the access controls on a report, by:
  - i) authenticating the security principal; and
  - ii) applying the security filter to the report.

22. The method of claim 2 further comprising the steps of:

- a) defining a security principal;
- b) creating an access control entry for at least one element of information in the data store, which access control entry grants or denies access to the security principal;
- 5 c) defining a security filter on a view, by:
  - i) creating a filter that identifies at least one row in the view; and
  - ii) creating an access control entry denying the security principal access to any information to be excluded by the filter; and
- d) enforcing the access controls on a report, by:
  - i) authenticating the security principal; and
  - ii) applying the security filter to the report.

10 23. The method of claim 1 further comprising the steps of:

- a) creating a reusable list of keys or row identifiers by specifying a filter on a view and selecting at least one of the resulting keys or rows; and
- 15 b) using the list as a set of filter values in conjunction with an IN or NOT IN filter operator.

24. The method of claim 2 further comprising the steps of:

- a) creating a reusable list of keys or row identifiers by specifying a filter on a view and selecting at least one of the resulting keys or rows; and
- 20 b) using the list as a set of filter values in conjunction with an IN or NOT IN filter operator.

25. The method of claim 2 wherein the report field includes two or more non-consecutive references to a particular view.

26. A computer system for building a report, the system comprising:

- a) means for defining a relational abstraction of a data store, the definition including a plurality of views, fields associated with the views, and any relations between the views;
- 5 b) means for selecting one view as a base view of the report; and
- c) means for creating at least one report field using the views, fields and relations associated with the base view.

27. The system of claim 26 further comprising a means to create the report field by providing means for:

- 10 a) specifying a relation path from the base view to a predetermined destination view, the relation path including zero or more relations;
- b) creating an aggregate field where the selected relation path contains any to-many relations, by:
  - i) creating at least one nested field to be aggregated, where the relation path of the nested field begins at the destination view of the aggregate field; and
  - 15 ii) specifying an expression for aggregating the values associated with the nested field; and
- c) creating a scalar field where the relation path does not contain any to-many relations by:
  - i) referencing a source field from the destination view, or
  - 20 ii) creating an expression field that includes at least one nested field, where the relation path of the nested field begins at the destination view of the expression field.